Balance prediction of the inertia wheel pendulum by using swing up and PID controller

Hilmi Kuşçu[1] and Eray Yılmazlar[2]
Trakya University, Mechanical Engineering Department, Edirne, Turkey[1]
Kırklareli University, Technical Sciences Vocational School, Kırklareli, Turkey[2]
hilmi@trakya.edu.tr-eray.yilmazlar@klu.edu.tr

Abstract: *In this paper, inertia wheel pendulum balance control is performed by using swing up and PID controller. Paper provides predictions on real time design balance system. Predictions were performed through data that were classified and tested by machine learning via MATLAB. Data obtained a result of the analyze of balance positions and swinging times of the wheel different diameters and weights in real-time. Through to this work will be able to predictable which wheel characteristics can be controlled and balanced.*

Keywords: *MACHINE LEARNING, PID CONTROLLER, WHEEL PENDULUM.*

I. Introduction

Inertia wheel pendulum (IWP) is a nonlinear and underactuated system with two degrees of freedom. The pendulum structure consists of a pendulum rod that can swing freely in the vertical axis, a rotating wheel in the same axis with the rod, and a motor that produces a rotational movement[1]. The main purpose of the IWP systems is the alignment of the pendulum wheel on the vertical axis. Balancing is the process of raising and aligning the pendulum with the control methods of the torque produced by the DC motor.

Machine learning is an artificial intelligence field that enables the system to create a model by using learning from past experiences and to make estimation against future situations[2]. Machine learning is used in many disciplines in our age. It provides convenience to devices and people in data analysis, decision making, estimation, conclusion and classification processes. The combination of machine learning and artificial intelligence with devices has enabled the creation of smart, self-guessing capable devices. Today, many systems are used by making use of the capabilities of artificial intelligence. These abilities were used in this study to estimate the balance of the balancing system.

The aim of this study is to control the pendulum wheel in different weights and wheel diameters. In addition, according to the weight and diameter variables to determine the ideal range for the balance of the pendulum is done by machine learning algorithms.

The studies on IWP started in 2001 and continue with many types of control methods and designs[3,4,5]. When the studies on IWP were examined, Hernández controlled the IWP system with PI in 2003 [6]. Victor carried out IWP balancing with limited torque technique in 2005 [7]. Victor made the dynamics and control of the IWP system in 2018 [8]. Jafar controlled the double pendulum mechanism with PID [9].

In this study, 39 different experiments were conducted and balance condition was analyzed together with disturbing factors affecting the system. Rest of the information of this document is organized as follows: Sec. A is devoted to describing IWP system modeling and dynamics. Sec. B indicates the system design procedure and control methods. In Sec. C, data analysis of wheel balance and make a prediction, classification using machine learning application. Then the final section reveals the results of this study.

1.1. Wheel Pendulum System and Dynamic Models

The IWP system consists of three parts. These parts are pendulum rod, pendulum wheel, and dc motor. In the IWP control design, the dynamic model of the system is calculated by the Euler-Lagrange formula 1: Euler-Lagrange Equations ($\mathcal{L}$) is a very useful method of extracting the equations of motion of the dynamic system. For the solution of the Euler-Lagrange equation, firstly there must be a difference in kinetic energy and potential energy [10,11] .

Table 1 *Wheel Characteristics And Balance data*

Parameter	Description	Units
m_w	Wheel mass	kg
m_P	Pendulum mass	kg
m_m	DC motor mass	kg
J_w	Wheel moment of inertia	$Kg.m^2$
J_p	Pendulum moment of inertia	$Kg.m^2$
J_m	DC motor moment of inertia	$Kg.m^2$
φ	Rotation angle of wheel	rad
θ	Tilted angle of the pole	rad
r	Wheel radius	m
l	Pendulum rod length	m
g	Gravity acceleration	m/s^2

$$\mathcal{L} = \mathrm{Ke} - \mathrm{Pe} \tag{1}$$

Ke: Total kinetic energy of system

Pe : Total potential energy of system

The total kinetic energy consists of the wheel, the wheel bar and the kinetic energy of the engine. kinetic energy of wheel, kinetic energy of pendulum rod, kinetic energy of motor and total kinetic energy equations describe in (2).

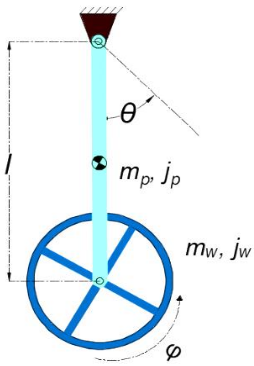

Fig. 1 *Parameter of IWP configuration*

To simplify the equation when defined as A in eq. 2-3:

$$A = J_w + J_p + J_m + m_w l^2 + m_p \left(\frac{l}{2}\right)^2 + m_m l^2 \tag{2}$$

$$Ke = \frac{1}{2} A\dot{\theta}^2 + \frac{1}{2} J_w \dot{\varphi}^2 \tag{3}$$

The total potential energy of the system appears in eq. 4:

$$Pe = \left(m_w l + m_m l + m_p \frac{l}{2}\right) g \cos\theta \tag{4}$$

To simplify the equation when defined as U in eq. 5-6:

$$U = m_w l + m_m l + m_p \frac{l}{2} \tag{5}$$

$$Pe = Ug\cos\theta \tag{6}$$

The Lagrange difference equation appears eq. 7:

$$\mathcal{L} = \frac{1}{2}A\dot{\theta}^2 + \frac{1}{2}J_w\dot{\varphi}^2 - Ug\cos\theta \tag{7}$$

When the difference equation is written in the general Lagrange expression in 8: and q1=θ accepted q2=φ, the equation is determined.

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{q}_i}\right) - \left(\frac{\partial\mathcal{L}}{\partial q_i}\right) = \tau_i \tag{8}$$

When differential equation solutions are made, eq. 9-10: is found.

$$A\ddot{\theta} + J_w\ddot{\varphi} - Ugsin\theta = 0 \tag{9}$$

$$J_w\ddot{\theta} + J_w\ddot{\varphi} = \tau \tag{10}$$

From these eq. the mathematical model of the system is determined in eq. 11:

$$\begin{bmatrix} A & J_w \\ J_w & J_w \end{bmatrix}\begin{bmatrix} \ddot{\theta} \\ \ddot{\varphi} \end{bmatrix} + \begin{bmatrix} -Ugsin\theta \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} \tag{11}$$

2. System Control Method

Different control methods are used at various stages in order to realize the movement of the pendulum from 0 degrees to 180 degrees with the least energy consumption. In the design of the pendulum, the movement is provided by DC motor with control signals generated by the Arduino control card as shown in the fig. 2 block diagram. During the swing process, the angle and position information are measured by the encoder and conveyed to the control unit for feedback. In this study, two different methods are used.

The first is the swing up control of the pendulum and the second is the balance control of the pendulum with PID.

The Swing up control does not balance the pendulum to the desired vertical alignment but supports it to arrive in the angular range where the balance will take place. The position of the pendulum wheel is 0° at the beginning. The ramp function or any triggering is applied to start the wheel swinging. As a result of the trigger, the wheel starts to swinging clockwise and counter clockwise. The swinging should be supported to increase the pendulum from 0° to 180° degrees. This support is applied with the torque produced by the dc motor. The support torque is applied when the variable pendulum angle value is maximum and the acceleration is zero during the swinging process. As a result of these processes, the pendulum is increased to the desired swinging range The pendulum control process switches to the balance control range when the swinging operation is complete[12].

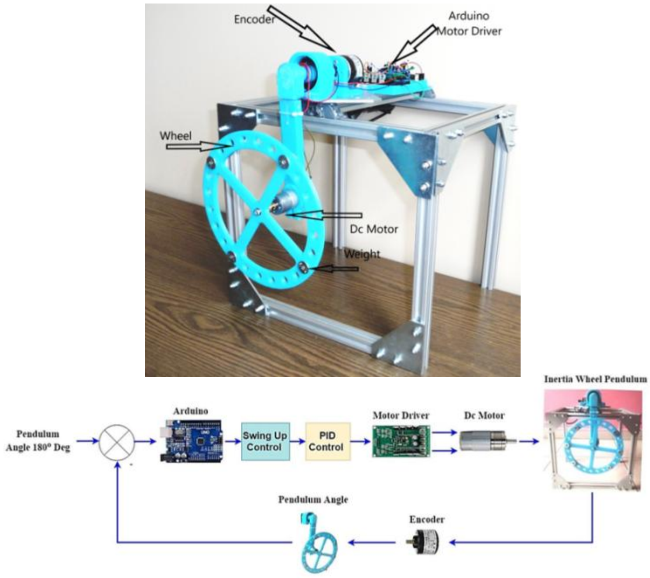

Fig. 2 Wheel pendulum mechanism and block diagram

Proportional-integral-derivative (PID) controllers are the most important control systems used to control processes, due to their simple and easy design, low cost and wide range of applications [13]. The main purpose of the PID control system is that the controlled process variable reaches the target in minimum time with minimum error difference. The PID control compares the reference value and feedback variables. In order to eliminate the error between two variables, proportional, integral and derivative parameters are applied to the system. These parameters modify according to the system model[14]. These Parameters are used in continuous cycling method and system response methods developed by Ziegler-Nichols. Large settling time and overshoot are minimized by Kp Ki Kd parameters.

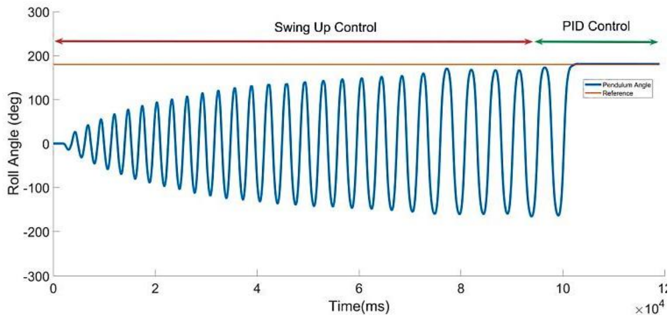

Fig. 3 Wheel pendulum mechanism and block diagram

3. Prediction Via Machine Learning

Machine learning software algorithms classify, handle and analyze the data in the system and, as a result, perform functions such as make decisions, prediction, and completion. As a result of the machine learning analysis, it increases the accuracy, precision and the value of efficiency by estimating according to similar input analyse data.

3.1 Data Collecting

In this study, 3 different diameter wheels were used in machine learning analysis. Each wheel is fixated with different weights. When the wheel pendulum project was running in different diameter and weight case, data collection operations were collected for machine learning by MATLAB-Arduino serial communication and observation data. As a result of the data collected, it has been measured whether the wheel has reached its balance position and how many swing periods have occurred to reach it.

Table 2. Sample of Wheel Characteristics And Balance data

Wheel Radius Wheel Intertia	Wheel Mass	Settling time	Balance
9R 550	90	50	1
9R 900	110	50	1
9R 1128	125	50	1
9R 1265	140	53	1
7.5R 666	85	77	1
7.5R 760	115	80	1
7.5R 1200	195	90	0
7.5R 1375	225	90	0
6R 474	73	90	0
6R 512	80	92	0
6R 611	100	94	0
6R 732	110	98	1

3.2 Training Algorithm

The data were analyzed by MATLAB classification learner and trained for machine learning. As a result of the training, the best accuracy rate was determined and classified with k-Nearest Neighborhood (KNN) algorithm [15].

Euclidean distance was used in the KNN algorithm. Euclidean distance can be explained as the linear distance between two points in the classification process. x = {x1, x2, .., xn} and y = {y1, y2, .., yn} are used by handle the euclidean distance (d) eq. 13: between two points [16]. As a result of machine learning training, confusion matrix and ROC curve appear in the figure 4. True positive and true negative values are over %90. The accuracy is calculated as 92% in the ROC curve.

$$d = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{13}$$

The accuracy, recall, precision and f-measure of the classification process were calculated to determine the true accuracy rate of the prediction system [17].

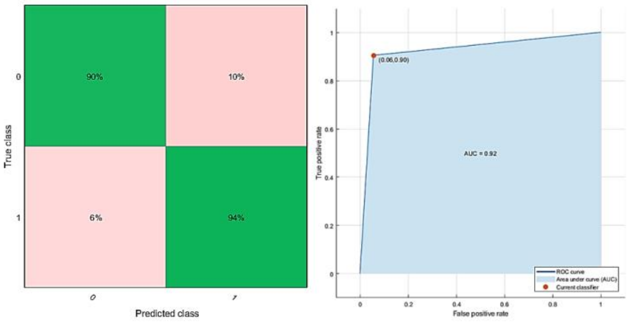

Fig. 4 *Confusion matrix and ROC curve*

False negative (FN) =0.10
False positive (FP) = 0.06
True negative (TN) = 0.94
True positive (TP) =0.90

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} = \frac{0.90 + 0.94}{0.90 + 094 + 0.06 + 0.10} = 1.58 \tag{14}$$

$$\text{Recall} = \frac{T_P}{T_P + F_N} = \frac{0.90}{0.90 + 0.10} = 0.90 \tag{15}$$

$$\text{Precision} = \frac{T_P}{T_P + F_P} = \frac{0.90}{0.90 + 0.06} = 0.9375 \tag{16}$$

$$\text{F} - \text{measure} = \frac{2 * \text{recall} * \text{precision}}{\text{recall} + \text{precision}} = \frac{2 * 0.90 * 0.9375}{0.90 + 0.9375} = 0.918 \tag{17}$$

As a result of calculations, the actual success rate in the predict process was found to be 91.8 percent.

3.3 Test and Control Operation

The accuracy of the KNN algorithm is tested with values that are different from the training data shown in the table 3. The accuracy of machine learning was 83.33% compared to the predicted rate and the actual values.

Table 3. *Test Data and Balance Predict*

Wheel Radius Wheel Intertia	Wheel Mass	Predict of Balance	Real of Balance	Acc.
4R-350	80	0	0	%100
5R-400	60	0	0	%100
6R-750	140	1	1	%100
7.5R-1025	100	1	1	%100
7.5R-750	100	1	1	%100
7.5R-1400	185	0	0	%100
8R-1300	140	1	1	%100
8R-1400	200	0	0	%100
9R-1000	150	1	1	%100
9R-1200	130	1	1	%100
9R-1750	180	1	0	%0
10R-2000	250	0	0	%100
11R-2000	150	1	1	%100
12R-2250	170	0	1	%0
Total Accuracy				%83.33

4. Conclusion

In this study a control of nonlinear and underactuated system was achieved by swing up control and PID control at various angle stages. The most significant factors affecting the stability of IWP systems are wheel diameter and wheel weight. These inputs were applied to the IWP system with different values. In the control process, it was observed that the weight supported to balance position until the to an amount. In case the pendulum weight is light or too heavy, the balancing operation was not realized. As the wheel radius expands, the pendulum was more easily balanced with lighter weight in the process.

The novelty of this study unlike the other IWP studies is that the wheel parameters where the balance position takes place is trained by machine learning algorithm and predicts the balance position at different wheel types. When the machine learning balance estimate and the real balance position of the IWP were compared, it was found that the similarity was 83.33%.

As a result of this study, it can be predicted whether the IWP system is stable for the balance position according to the input parameter characteristics using machine learning In the case of the predicted result of the IWP system is unbalanced, it will be determined that different wheel parameters should be applied for balance. In addition to the model dynamic calculations, the balance state of the IWP system will be determined more accurately. IWP applications will be more realistic because all the factors affecting the balancing process will be taken into account. In the continuation of this study, it is aimed that the input information will be entered into the control card via the interface screen and evaluated in real time with the machine learning.

5. References

[1] C. Aguilar-Avelar, R. Rodríguez-Calderón, S. Puga-Guzmán, and J. Moreno-Valenzuela, “Effects of nonlinear friction compensation in the inertia wheel pendulum,” *J. Mech. Sci. Technol.*, vol. 31, no. 9, pp. 4425–4433, 2017, doi: 10.1007/s12206-017-0843-4.

[2] B. Leo, “Bagging predictors,” *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[3] M. W. Spong, P. Corke, and R. Lozano, “Nonlinear control of the Reaction Wheel Pendulum,” *Automatica*, 2001, doi: 10.1016/S0005-1098(01)00145-5.

[4] O. D. Montoya, V. M. Garrido, W. Gil-gonzález, and C. Orozco-henao, “Passivity-Based Control Applied of a Reaction Wheel Pendulum : an IDA-PBC Approach Passivity-Based Control Applied of a Reaction Wheel Pendulum : an IDA-PBC Approach,” *2019 IEEE Int.*

Autumn Meet. Power, Electron. Comput., pp. 1–6, 2019.

[5] I. Siradjuddin, E. R. K. Pradani, E. Rohadi, S. Adhisuwignjo, M. Kusumawardani, and I. M. Fitriani, "Designing, implementing and analysing optimal controllers on a non-linear reaction wheel pendulum," *J. Phys. Conf. Ser.*, vol. 1402, no. 4, 2019, doi: 10.1088/1742-6596/1402/4/044025.

[6] V. M. Hernández and H. Sira-Ramírez, "Generalized PI Control for Swinging up and Balancing the Inertia Wheel Pendulum," *Proc. Am. Control Conf.*, vol. 4, no. 2508, pp. 2809–2814, 2003, doi: 10.1109/acc.2003.1243748.

[7] V. Santibañez, R. Kelly, and J. Sandoval, "Control of the inertia wheel pendulum by bounded torques," *Proc. 44th IEEE Conf. Decis. Control. Eur. Control Conf. CDC-ECC '05*, vol. 2005, no. 3, pp. 8266–8270, 2005, doi: 10.1109/CDC.2005.1583500.

[8] N. Kant and R. Mukherjee, "Impulsive Dynamics and Control of the Inertia-Wheel Pendulum," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3208–3215, 2018, doi: 10.1109/LRA.2018.2851029.

[9] H. I. Jaafar *et al.*, "Efficient control of a nonlinear double-pendulum overhead crane with sensorless payload motion using an improved PSO-tuned PID controller," *JVC/Journal Vib. Control*, vol. 25, no. 4, pp. 907–921, 2019, doi: 10.1177/1077546318804319.

[10] R. Ortega and P. J. Nicklasson, *Passivity-based Control of Euler-Lagrange Systems :* .

[11] A. Türkmen, M. Y. Korkut, M. Erdem, Ö. Gönül, and V. Sezer, "Design, implementation and control of dual axis self balancing inverted pendulum using reaction wheels," *2017 10th Int. Conf. Electr. Electron. Eng. ELECO 2017*, vol. 2018-Janua, no. 1, pp. 717–721, 2018.

[12] M. W. Spong, "The Swing Up Control of the Acrobot," *IEEE Control Syst. Mag.*, vol. 15, no. May 1994, pp. 49–55, 1995.

[13] C. Sánchez-López, V. H. Carbajal-Gómez, M. A. Carrasco-Aguilar, and F. E. Morales-López, "PID controller design based on memductor," *AEU - Int. J. Electron. Commun.*, 2019, doi: 10.1016/j.aeue.2019.01.019.

[14] M. Tuna, C. B. Fidan, S. Kocabey, and S. Görgülü, "Effective and reliable speed control of permanent magnet DC (PMDC) motor under variable loads," *J. Electr. Eng. Technol.*, vol. 10, no. 5, pp. 2170–2178, 2015, doi: 10.5370/JEET.2015.10.5.2170.

[15] K. Shi, L. Li, H. Liu, J. He, N. Zhang, and W. Song, "An improved KNN text classification algorithm based on density," *CCIS2011 - Proc. 2011 IEEE Int. Conf. Cloud Comput. Intell. Syst.*, vol. 4, no. 3, pp. 113–117, 2011, doi: 10.1109/CCIS.2011.6045043.

[16] L. K. S. Kilian Q. Weinberger, "Distance Metric Learning for Large Margin Nearest Neighbor Classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, 2009.

[17] S. Kocaoğlu and E. Akdoğan, "Design and development of an intelligent biomechatronic tumor prosthesis," *Biocybern. Biomed. Eng.*, 2019, doi: 10.1016/j.bbe.2019.05.004.